Équivalence du Consensus et de la Diffusion dans les Réseaux à Omissions Bornées

Emmanuel Godard[1] and Joseph Peters[2]

[1] *LIF, Université Aix-Marseille&CNRS, France*
[2] *School of Computing Science, Simon Fraser University, Canada*

Nous comparons l'existence de solutions pour le problème du Consensus ou le problème de la Diffusion dans le cadre de réseaux de communication synchrones où la transmission de message n'est pas fiable. Certains messages peuvent être perdus et à chaque ronde le nombre de messages perdus est borné d'une certaine manière. Nous montrons que dans ce cas, *et quelle que soit la manière de compter les pertes (localement, globalement,...)* le problème du Consensus est équivalent au problème de la Diffusion tant en terme de calculabilité qu'en terme de complexité.

Keywords: consensus, diffusion, réseaux synchrones, tolérance aux pannes, pertes de messages

Les problèmes du Consensus et de la Diffusion sont deux problèmes fondamentaux de l'algorithmique distribuée. On s'intéresse ici à leurs solutions éventuelles dans le cadre d'un réseau synchrone où certains messages peuvent être perdus (omissions). A chaque ronde les combinaisons de pertes possibles sont les mêmes (modèle des fautes mobiles [SW07]). On montre que si ces combinaisons sont obtenues en bornant le nombre de fautes, *quelle que soit la manière de les compter*, alors le Consensus et la Diffusion sont des problèmes équivalents. Ce résultat permet notamment de retrouver des résultats dans le cas d'une borne globale [FG11] (répondant à une question ouverte de [SW07]) ainsi que dans celui d'une borne locale [SWK09]. Cette présentation est une version autonome d'un des résultats de [GP11].

1 Définitions et Notations

Réseaux de Communication Un réseau de communication est représenté par un graphe dirigé $G = (V,E)$ (pas nécessairement symétrique). Tous les nœuds possèdent un identifiant unique. La communication est port-à-port. Tous les sous-graphes que nous considérons ici seront des sous-graphes couvrants. Nous identifierons donc l'ensemble d'arcs avec le sous-graphe couvrant correspondant.

Schémas d'Omission Dans notre modèle, la communication est synchrone mais non-fiable. Lorsqu'une omission (perte de message) se produit au cours d'une ronde, la communication correspondante sera représentée par le sous-graphe couvrant de G contenant exactement les arcs sans omissions lors de cette ronde. Les arcs absents représentent les pertes. Le graphe sous-jacent $G = (V,E)$ étant donné, on note $O_G = \{(V,F) \mid F \subseteq E\}$. Cet ensemble représente toutes les communications simultanées possibles.

Définition 1.1. *Un élément de O_G est appelé* évènement de communication *(ou simplement* évènement*). Un* scénario à omissions *(ou plus simplement* scénario*) est une séquence infinie d'évènements de communication. Un* schéma d'omission *sur G est un sous-ensemble $S \subset O_G$. Un réseau* soumis à un schéma d'omission *S est un réseau où les scénarios possibles sont toutes les séquences constituées d'éléments de S.*

Nombre d'Omissions Borné On définit ici les notions de métrique d'omissions et d'omissions bornées.

Définition 1.2. *La fonction $\mu : O_G \longrightarrow \mathbb{N}$ est une* métrique d'omissions *sur G si*

(fiabilité de G) $\mu(G) = 0$

(monotonie) $\forall H,K \in O_G,\ H \subseteq K \Longrightarrow \mu(H) \geq \mu(K)$

Intuitivement, s'il y a moins d'arcs dans H, son nombre d'omissions, *quelle que soit la manière de les compter*, est au moins égal à celui de K. De tels exemples de métriques sont la métrique globale $\mu_g(F) =$

$\#(E \backslash F)$ ou encore la métrique locale $\mu_\ell(F) = \max_{u \in V} \#\{v \in V \mid (u,v) \in E \backslash F\}$.

Un schéma d'omission S est *définissable par borne* (ou plus simplement *borné*) s'il existe une métrique d'omissions μ et $k \in \mathbb{N}$ tels que $S = \{H \in O_G \mid \mu(H) \leq k\}$. En remarquant que $\mu(G) = 0$ dans tous les cas, on a toujours $G \in S$.

Proposition 1.3. *Soit S un schéma d'omission borné (de métrique μ et de borne $k \in \mathbb{N}$).*
Pour tout $H, H' \in S$ et tout $a \in H'$, $H \cup \{a\} \in S$. En particulier $H \cup H' \in S$.

Démonstration. Le graphe H' est un sous-graphe couvrant donc $a \in E$. Donc $H \cup \{a\} \subseteq E$. En utilisant la propriété de monotonie, on obtient que $\mu(H \cup \{a\}) \leq \mu(H)$. Or $\mu(H) \leq k$, donc $H \cup \{a\} \in S$. □

Nous présentons ensuite deux exemples de système avec deux processus : $G = \{\circ \leftrightarrows \bullet\}$.

Exemple 1. *Le schéma d'omission $O^1 = \{\circ \leftrightarrows \bullet, \circ \leftarrow \bullet, \circ \rightarrow \bullet\}$ a été très étudié et correspond à la situation où il se produit globalement au plus une omission par ronde. Il est définissable par borne (avec μ_g et $k = 1$).*

Exemple 2. *Le schéma d'omission $\mathcal{H} = \{\circ \leftarrow \bullet, \circ \rightarrow \bullet\}$ décrit un système dans lequel au plus un message peut être transmis simultanément. En particulier, si un seul message est envoyé, il peut ne pas être reçu.*
Ce schéma n'est pas définissable par borne car le graphe sous-jacent $\circ \leftrightarrows \bullet$ n'appartient pas à $\mathcal{H}$.

Exécution d'un Algorithme Distribué soumis à Omissions Considérons un processus u et l'un de ses voisins sortants v. Un message *msg* est envoyé depuis u vers v, selon les instructions de l'algorithme $\mathcal{A}$. Supposons que l'évènement $H \in O_G$ se produise. La fonction de réception $recv(u)$ renverra *msg* seulement si $(u,v) \in H$. Sinon, la valeur renvoyée sera *null*. Les messages envoyés lors d'une ronde peuvent être reçus uniquement dans cette ronde. Une *exécution de $\mathcal{A}$ soumise à S* est la séquence (potentiellement infinie) de tels échanges de messages et configurations correspondantes lorsque les évènements appartiennent à S.

Définition 1.4. *Soit S un schéma d'omission, un algorithme $\mathcal{A}$* résout un problème $\mathcal{P}$ de manière $S-$fiable *avec la configuration initiale ι, si, pour tout scénario w soumis à S, il existe un préfixe fini w' de w tel que l'*état *$s^u(w')$ de chaque processus $u \in V$ satisfait les spécifications de $\mathcal{P}$ pour la configuration initiale ι.*

Diffusabilité Soit H un évènement et un nœud u de V. Un nœud $v \in V$ est *accessible depuis u dans H* si il existe un chemin dirigé de u vers v dans H. Le nœud u est une *source* de H si tout $v \in V$ est accessible depuis u dans H. Un schéma d'omission S est *source-incompatible* si il n'existe aucune source commune à tous les éléments de S. On rappelle que le problème de la Diffusion consiste à trouver un sommet v et un algorithme permettant de diffuser depuis v. Avec ces définitions, on a clairement (cf. la preuve dans [GP11]) :

Théorème 1.5. *Soit G un graphe et S un schéma d'omission pour G. Il est possible de diffuser de manière $S-$fiable, si et seulement si S n'est pas source-incompatible.*

2 Équivalence du Consensus et de la Diffusion

Nous allons prouver que les Problèmes du Consensus et de la Diffusion sont équivalents dans la large (et peut-être la seule comptant en pratique) famille des schémas d'omissions définissables par borne. Les conséquences sont très importantes car vérifier la Diffusabilité est très simple (voir Théorème 1.5). Le problème du Consensus est le suivant. Chaque nœud possède une valeur initiale 0 ou 1 et doit décider une valeur identique aux autres nœuds de telle manière que si les valeurs initiales sont déjà identiques, c'est cette valeur commune qui est décidée. Il est immédiat que si le problème de la Diffusion possède une solution depuis un nœud v, alors il existe un algorithme pour le Consensus en diffusant la valeur initiale de v.

Théorème 2.1 (Équivalence). *Soit $S \subset O_G$ un schéma d'omission borné sur un graphe G. Alors le Consensus admet une solution $S-$fiable si et seulement si la Diffusion admet une solution $S-$fiable.*

Démonstration. Par Théorème 1.5, il suffit de montrer que si S est source-incompatible alors le Consensus ne peut être résolu. La preuve utilise la technique classique de bivalence.

On suppose qu'il existe un algorithme $\mathcal{A}$ résolvant le Consensus. On dit qu'une configuration est $0-$valente (resp. $1-$valente) si toute exécution de $\mathcal{A}$ depuis cette configuration aboutit à une valeur de décision 0 (resp. 1). Une configuration est univalente si elle est 0 ou $1-$valente. Elle est bivalente sinon.

Initialisation bivalente. Soit $V = \{v_1, \ldots, v_n\}$ et soit ι_ℓ la configuration initiale dans laquelle v_i a pour valeur initiale 0 si $i > \ell$, et 1 sinon. Comme ι_0 (resp. ι_n) doit être $0-$valent (resp. $1-$valent) par définition du Consensus, si toutes les configurations ι_ℓ, $1 \leq \ell \leq n$ sont univalentes, alors il existe ℓ tel que $\iota_{\ell-1}$ est $0-$valent et ι_ℓ est $1-$valent. Comme S est source-incompatible, il existe $H \in S$ tel que v_ℓ ne soit pas source de H. Donc il existe un sommet u non-accessible depuis v_ℓ. En remarquant que ces initialisations $\iota_{\ell-1}$ et ι_ℓ ne diffèrent qu'en v_ℓ, les exécutions depuis ces deux configurations initiales où H est le seul évènement aboutissent à un état identique pour u. Et u doit décider la même valeur dans les deux exécutions. Ceci est en contradiction avec les valences de ι_0 et ι_n.

Extension bivalente. Nous montrons que si une exécution partielle produit une configuration bivalente alors il existe une extension de cette exécution qui est encore bivalente. On raisonne par l'absurde, supposons qu'aucune extension ne soit bivalente, par conséquent elles sont toutes univalentes et comme la configuration étendue était bivalente, il existe une extension $0-$valente (resp. $1-$valente). Notons H_0 (resp. H_1) l'évènement correspondant. D'après le Lemme 1.3, il existe $G_0, G_1, \ldots, G_q \in S$ tels que $G_0 = H_0$ et $G_q = H_1$ et tels que G_i et G_{i+1} ne diffère que d'un seul arc, pour $0 \leq i < q$. En effet, il est possible d'ajouter des arcs à H_0 pour obtenir $H_0 \cup H_1$, puis d'enlever des arcs pour obtenir H_1. Par hypothèse d'univalence, et comme précédemment, il existe $\ell < q$ tel que la configuration atteinte après G_ℓ, σ_0, soit $0-$valente et celle atteinte après $G_{\ell+1}$, σ_1, soit $1-$valente. Notons a l'arc différant entre G_ℓ et $G_{\ell+1}$, et v le sommet de destination de cet arc. Comme S est source-incompatible, il existe $H \in S$ tel que v ne soit pas source de H. Donc il existe un sommet u non-accessible depuis v. Par conséquent, si on étend l'exécution depuis σ_0 et σ_1 en n'ayant comme unique évènement H, on aura un état identique pour u. Donc u doit décider la même valeur dans les deux exécutions. Contradiction.

Conclusion. Par conséquent, à partir de la configuration initiale bivalente, il est possible de construire une exécution infinie qui soit toujours bivalente, ie tel que l'algorithme $\mathcal{A}$ ne soit jamais terminé. Ce qui contredit le fait que l'algorithme $\mathcal{A}$ résolve le Consensus. □

Soit $O_G^f(G)$ l'ensemble des sous-graphes couvrants de G avec au plus f arcs manquants par rapport à G. Une borne supérieure sur f au sujet de l'existence de solution au Consensus soumis à $O_G^f(G)$ est donné dans [SW07], et cette borne fut démontrée exacte par une technique de simulation *ad hoc* dans [FG11]. Ce résultat apparaît maintenant comme un corollaire immédiat du Théorème 2.1.

Corollaire 2.2. *Soit $f \in \mathbb{N}$ et G un graphe (symétrique). Le Consensus admet une solution soumis à $O_G^f(G)$ si et seulement si $f < c(G)$, où $c(G)$ est la connectivité du graphe G.*

Notons qu'il n'est pas possible de diffuser de manière $\mathcal{H}-$fiable (Exemple 2) alors que le Consensus admet une solution pour ce schéma d'omission. L'algorithme où la valeur décidée par un nœud est la valeur reçue, si elle est non *null*, sa propre valeur initiale sinon, résout le Consensus en une ronde. Par conséquent, les deux problèmes peuvent ne pas être équivalents dès que les omissions ne sont pas bornées. Lorsque les omissions sont bornées, l'équivalence concerne également le nombre de rondes nécessaires.

Proposition 2.3. *Soit $S \subset O_G$ un ensemble d'évènements définissable par borne sur un graphe G. Le problème du Consensus admet une solution $S-$fiable en r rondes si et seulement si le problème de la Diffusion admet une solution $S-$fiable en r rondes.*

Démonstration. Puisqu'un algorithme de Diffusion peut toujours être utilisé pour réaliser le Consensus, il suffit de montrer que le Consensus ne peut être résolu en moins de rondes que la Diffusion. On suppose qu'il existe un algorithme $\mathcal{A}$ résolvant le Consensus en r_c rondes et que la Diffusion nécessite au moins $r_d > r_c$ rondes. Ainsi, une Diffusion, de toute origine, aura un nombre de rondes strictement supérieur à r_c.

On va montrer qu'il doit y avoir une configuration initiale bivalente. Soit $V = \{v_1, \ldots, v_n\}$ et $\iota_\ell, 0 \leq \ell \leq n$, les configurations initiales définies dans la preuve précédente. Si toutes ces configurations sont univalentes, alors, il existe ℓ tel que $\iota_{\ell-1}$ est $0-$valent et ι_ℓ est $1-$valent. Comme une Diffusion initiée en v_ℓ nécessite plus que r_c rounds, alors il existe un sommet u qui n'a reçu aucune valeur de v_ℓ, donc aucune exécution de longueur r_c de l'algorithme $\mathcal{A}$ depuis les configurations initiales $\iota_{\ell-1}$ et ι_ℓ ne peut être distinguée d'une autre par u. En conséquence, u décide nécessairement la même valeur pour les deux configurations initiales : contradiction. Maintenant, on montre de la même manière que si toutes les extensions d'une configuration bivalente sont univalentes, alors l'algorithme $\mathcal{A}$ nécessite plus de r_c rondes supplémentaires. □

Il existe de très nombreux résultats concernant le problème de la diffusion dans des familles de réseaux spécifiques. Les graphes quelconques sont étudiés dans [CDP94] et les hypergraphes dans [MV98]. Un algorithme optimal pour la famille des hypercubes est donné dans [DV99]. Pour un hypercube de dimension n, si au plus $n-1$ messages sont perdus à chaque ronde, alors la Diffusion peut être réalisée en $n+2$ rounds, comparé à n rondes quand il n'y a aucune erreur d'omissions. Dans [DV04], l'impact précis sur la Diffusion du nombre exact de défaillances est donné. En utilisant directement les résultats de [DV04], on obtient :

Corollaire 2.4. *Dans un hypercube de dimension n, si le nombre global d'omissions est au plus f par ronde, alors*

1. *si $f \geq n$, alors le Consensus n'admet aucune solution,*
2. *si $f = n-1$, le Consensus admet une solution optimale en $n+2$ rondes,*
3. *si $f = n-2$, le Consensus admet une solution optimale en $n+1$ rondes,*
4. *si $f < n-2$, le Consensus admet une solution optimale en n rondes.*

L'exemple suivant montre que si le schéma de défaillance n'est pas borné, alors il est possible que le Consensus admette une solution avec strictement moins de rondes que la Diffusion.

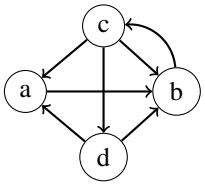

FIGURE 1: Évènement H_1.

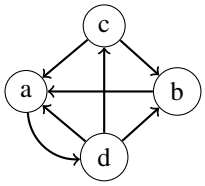

FIGURE 2: Évènement H_2.

Exemple 3. *Soit le schéma $\{H_1, H_2\}$ où H_1 (resp. H_2) est donné par Fig. 1 (resp. Fig. 2). Ce schéma n'est pas borné car le graphe $H_1 \cup H_2$ n'y appartient pas.*

Le sommet d a besoin de deux rondes pour diffuser soumis à $\{H_1\}$, et le sommet c a besoin également de deux rondes pour diffuser soumis à $\{H_2\}$. Les sommets a et b nécessitent plus que deux rondes.

Cependant, il existe un algorithme de Consensus qui décide en une seule ronde. Remarquons que chaque nœud peut détecter immédiatement lequel des deux évènements s'est effectivement produit. Par conséquent, l'algorithme dans lequel chaque nœud décide la valeur de c si H_1 se produit, ou bien celle de d si H_2 se produit, est un algorithme de Consensus.

Références

[CDP94] B. S Chlebus, K. Diks, and A. Pelc. Broadcasting in synchronous networks with dynamic faults. *NETWORKS*, 27 :309—318, 1994.

[DV99] Stefan Dobrev and Imrich Vrto. Optimal broadcasting in hypercubes with dynamic faults. *Information Processing Letters*, 71 :81—85, 1999.

[DV04] Stefan Dobrev and Imrich Vrto. Dynamic faults have small effect on broadcasting in hypercubes. *Discrete Applied Mathematics*, 137(2) :155–158, March 2004.

[FG11] Tristan Fevat and Emmanuel Godard. About minimal obstructions for the coordinated attack problem. In *Proc. of IEEE Int. Symp. on Parallel & Distributed Processing IPDPS'2011*, 2011.

[GP11] Emmanuel Godard and Joseph Peters. Consensus vs broadcast in communication networks with arbitrary mobile omission faults. In *Proc. of Sirocco'2011*, volume 6796 of *LNCS*, 2011.

[MV98] Gianluca De Marco and Ugo Vaccaro. Broadcasting in hypercubes and star graphs with dynamic faults. *Information Processing Letters*, 66(6) :321–326, 1998.

[SW07] Nicola Santoro and Peter Widmayer. Agreement in synchronous networks with ubiquitous faults. *Theor. Comput. Sci.*, 384(2-3) :232–249, 2007.

[SWK09] U. Schmid, B. Weiss, and I. Keidar. Impossibility results and lower bounds for consensus under link failures. *SIAM Journal on Computing*, 38(5) :1912–1951, 2009.